A Privacy Policy Framework for Grid and Web Services

Sarjeet Singh and Seema Bawa
University Institute of Engineering and Technology, Panjab University, Chandigarh-160014, India
Department of Computer Science and Engineering, Thapar Institute of Engineering and Technology, Patiala, India

Abstract: Privacy is becoming an important concern in service oriented environments like grid and web. Service providers and service requesters, both have complex set of privacy policies to better protect their interests. Both the parties need reality that the facts/information they reveal about themselves will not be used inappropriately by the other. Most of the past models available to address privacy requirements are ad-hoc, application specific or partially implemented. Some models address privacy concerns of either the requestor or the provider. This study proposes a generalized privacy model to handle the privacy requirements of both, the service providers and the service requesters. The model provides a uniform, integrated and platform neutral way to express, store, evaluate, enforce and manage privacy policies. The research also discusses a prototype implementation of the proposed privacy based authorization framework. The implementation has been done in NET environment with the support of WSE 3.0. The prototype implementation has shown that the model is able to meet the identified privacy requirements which suggest that the approach is workable and can be used to provide privacy based access to grid and web services.

Key words: Privacy concerns, service oriented environment, grid and web, privacy policies, privacy model, privacy based authorization framework

INTRODUCTION

Privacy is becoming a serious concern in service oriented environment like grid (Foster et al., 2001) and web (Booth et al., 2004), where a large number of users provide their personal information to service providers to gain access to their services. Service providers generally publish their privacy policies in plain English language to ensure customers that information provided by them will not be used inadequately. But the claims made by them (which are written in plain English) can’t give users the guarantee that their information will not be misused. The privacy model implemented by service providers must have the ability whereby users have control to check the misuse of their data as well as control over how their information can be collected, stored, used and shared. Besides handling privacy requirements of service requesters, privacy concerns of service providers should also be addressed, e.g., consider a secret service is provided by government to some of the officials of a department. If the officials have access to this secret service then they can leak information regarding security credentials required to access the service, address of the service, input and output from the service etc. Clearly, government in this case requires that officials must not know the credentials/attributes required to access the service, the actual address of the service and the input and output requirements of the service. Most of the privacy models address privacy requirements of service requesters and not of service providers. The scenario just described above demand a privacy model to handle privacy requirements of service providers. In addition to providing privacy features to service requesters, privacy model must have the ability to handle privacy requirements of service providers.

Besides the assurance of proper privacy of personal data, customers generally expect more privacy features from service providers. Sometimes customers want to hide their actual identity while accessing a service, e.g., consider an online system that provides e-voting services. Here individuals may wish to hide their identity while voting for a particular party but not the fact that they are accessing a service. Sometimes users want to hide even from the service provider that they have accessed a particular service, e.g., a user may want to hide from service provider that he has accessed a service.

Corresponding Author: Sarjeet Singh, University Institute of Engineering and Technology, Panjab University, Chandigarh-160014, India Tel: 91-9815951674
to purchase a video titled xyz, because, if service provider know that a particular user is accessing a particular service with a particular parameters then he can observe user’s usage pattern and make interpretations (data mining). So some support should be there to handle these types of privacy requirements. Though some of these issues can be resolved by granting anonymous access to services but this does not give a complete solution, e.g., if service providers does not know about user preferences then personalized access to services can not be provided. This is likely to disadvantage users as they will not be able to customize services according to their interests and hence will not experience quality of service. In fact, participation in the real world requires disclosure of information to some extent (Brar and Kay, 2005). To get customized access to services according to our preferences, it is necessary to supply preference details (private information) to service providers.

Thus, a Privacy Model should address a wider notion of privacy that focuses on providing users with the purpose of data collection, choice regarding collection and conditions under which data can be used so that users can control and make decisions regarding the disclosure of their private information. To achieve this, access control mechanisms implemented by authorization framework need to be extended to include privacy based access to data and services. The access control decisions should not be determined by the user’s identity alone, but by the task the individual user is currently performing and his conformance to established privacy policies. Personal data should be accessible to a user only if such an access is necessary to perform his task and if he is authorized to perform this task (Brar and Kay, 2005; Fischer-Hubner and Ott, 1998; Karjoh and Schunter, 2002; Mbanaso et al., 2006).

Most of the softwares available to construct grid and web services do not provide effective privacy models to address complex privacy requirements of service providers and service requesters. Much of the work done in this area is limited, e.g., Shibboleth (2005. http://shibboleth.internet2.edu/docs/internet2-mace-shibboleth-arch-protocols-200509.pdf) is an attempt to address privacy in an authorization environment but it is primarily focused on using pseudonymity. It does not provide a complete privacy based access control environment. Similarly the efforts like PRIMA (Lorch et al., 2003) and PERMIS (Chadwick et al., 2003) mainly provide policy based authorization framework but provide little support to address complex privacy related issues. The most commonly used toolkit to construct grid services GT4 also does not provide a comprehensive privacy model. The support is also missing in Alchemi (Luther et al., 2003) which is used to construct grid systems in .NET environment. Another effort, P3P, is a XML document that describes the data collection practices for a site but it does not provide technical mechanisms to check a given access request against the stated privacy policy. To our knowledge, no software is available to handle complex privacy requirements between service providers and service requesters while developing grid and web services. In this study we have made an attempt to define a general privacy model that is able to address the privacy issues of both, the service requesters and service providers. Its integration with the authorization framework is also discussed.

**PRIVACY MODEL**

A Privacy Model can be defined as a system that allow service requesters and service providers to state, evaluate and enforce their privacy requirements. It is a detailed description of all aspects of a system that relate to privacy. A privacy based authorization system grants specific type of access to specific requesters based on their authentication, what services/resources they are accessing, current state of the system and their conformance to established privacy policies. Privacy Model can be integrated with the Authorization Framework to provide privacy based access to services. This research presents a Privacy Model and its integration with Authorization Framework. In order to understand the architecture well, we have identified and defined the following elements:

**Elements of privacy model**

**Subject (SU):** Subject is an entity that wants to access services provided by service providers. Subject can be a user or any other entity on behalf of that user. Subjects send some information (which may be public or private) to service providers to access their services.

**Service (SR):** Service is a piece of software that provides some functionality and can be accessed by Subjects or other Services. Services are exposed in the environment along with their associated privacy policies. Services are provided by different service providers.

**Domain (DO):** Domain refers to a particular set of Subjects and Services. Subjects and Services of a Domain form collaboration. There exist complex privacy relationships between Subjects and Services of same/different Domains.
Service Provider (SP): SP is a physical organization/institution that exposes services/resources in a Domain. Services/Resources in a Domain may come from same or different Service Providers. Thus a Domain is a more dynamic entity than individual physical organization/institution.

Resource (R): Resource is an object that is accessed by Subjects. It is a sharable entity like data, record, file, databases, software etc. Resources are accessed through Services. In other words, a Resource is a Service. There are two types of resources:

Subject’s resources: These are the resources which are provided by Subjects to Service Providers, e.g., Subject’s name, date of birth, his private telephone number etc.

Service provider’s resources: These are the resources which are exposed by Service Providers to Subjects. Subjects can access these Resources based on their authorization status and their conformance to established privacy policies.

Privacy Policy (PP): Privacy Policy refers to the set of privacy rules/requirements associated with a Service/Subject. A Subject must conform to Privacy Policy associated with the Service in order to access that Service and a Service must respect the privacy policies of a Subject if it exposes Subject’s data to other entities.

Access (AC): Access is an operation that a Subject/Service performs on other Service. The access is provided based on conformance to established privacy and other security policies that are associated with that Service.

Privacy Index (PI): PI indicates the privacy level of Resources. Each Resource is associated with a PI value. Higher values of PI indicate more privacy. Private data/information is marked with higher values of PI. PI can take following values:

<table>
<thead>
<tr>
<th>PI</th>
<th>Type</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No privacy</td>
<td>Resource can be used anyway.</td>
</tr>
<tr>
<td>1</td>
<td>Partial privacy</td>
<td>Resource can be used with permission only.</td>
</tr>
<tr>
<td>2</td>
<td>Time limited privacy</td>
<td>Resource can be used with permission but for a limited time period.</td>
</tr>
<tr>
<td>3</td>
<td>Full privacy</td>
<td>Resource can not be used under any circumstances.</td>
</tr>
</tbody>
</table>

Purpose (PU): Tells why Subject’s Resources (e.g., his private/public information) are required by Service Provider and how these will be used. A Service Provider may require Subject’s Resources for marketing/research/data mining/to provide customized access to Services etc.

Actions (ACT): Tell what operations can be performed on Resources (i.e., data/information) provided by Subjects. The operations can be read/write/execute/copy/share/distribute/update/delete/append/view etc.

Conditions (CO): Conditions are privacy statements that describe some prerequisites which must be satisfied before Access to Resources is granted, e.g., a Condition may state that Subject’s age must be greater than 18 to access a Service/Resource.

Obligations (OB): Obligations are activities that must be performed by Service Providers after accessing/ providing access to Subject’s Resources. Example of Obligation can be to send a notification e-mail to parents if their children access a particular Service.

Consent (C): Consent is the permission given by Subjects, who provide their Resources, to explicitly state that their Resources can be used for a particular Purpose, e.g., a Subject may explicitly state that his telephone number can be used by marketing people to inform him about new schemes.

Privacy Controller (PC): PC is an entity that checks misuse of Subject’s Resources at Service Provider’s end i.e. it works for Subjects interests. It makes sure for Subjects that Service Providers are respecting Subject’s privacy requirements and are accessing/providing access to Subject’s Resources only for the purpose for which they have been sent.

Trusted Third Party (TTP): TTP is an independent entity that is trusted by both, Subjects (service requesters) and Service Providers. It can be used by Subjects to obtain signed credentials which they can later provide to Service Providers to obtain anonymous access to services. TTP can also be used by Service Providers to hide service details from Subjects. TTP plays an important role in handling Service Provider’s and service requester’s privacy requirements. TTP can be run by a government agency in order to make Service Providers and requesters to trust it. PC can also be a part of TTP.

Privacy infrastructure: Figure 1 shows a Distributed Environment, which can be grid or web, consisting of three Domains: Domain A, Domain B and Domain C. In
Fig. 1: Schematic representing distributed environment consisting of three domains showing how privacy issues among subjects and services of different domains are handled.

Now consider that SU-1's private information is required by Subject of some other Domain, say SU-2 of Domain B. Further consider that SR-2 is a Service of Domain C that provides access to SU-1's private information. Now accesses of SR-2 to SU-2 is provided only if this information is required by SU-2 to perform his task and the purpose of his task matches with the purpose for which this information was sent by SU-1. If the purpose for which SU-2 require SU-1's private information does not matches with the purpose for which SU-1 provided this information then access is denied. A notification regarding the use of SU-1's private information is sent to SU-1, if he has asked for it. The violation of any established Privacy Policy can be checked by Privacy Controller (PC). Though PC plays an important role in handling service requesters’ privacy requirements, it cannot be used by Service providers to address their privacy requirements. To handle service Provider’s privacy requirements, TTP is used. Following we will discuss how TTP can be used to address Service Provider’s privacy requirements.

Consider that Service Provider wants to expose a Service only to a selected set of Subjects and also wants to hide the security requirements associated with that Service (i.e., the security credentials/attributes required to access the Service). In order to provide this privacy
feature, we propose Service Provider to make use of TTP. Service providers share authentication information of Subjects and security requirements of Services with TTP. To access a hidden/secret service, Subjects pass their normal security credentials and attributes to TTP. TTP authenticate Subjects (as service providers share authentication information with TTP) and if authenticated, based on the credentials/attributes provided by Subjects, TTP generate a custom security token. In custom security token, only those credentials and attributes are included which are necessary to access the Service. This security token is then sent to Subjects. Only this security token can be used by Subjects to see and access hidden/secret Services. If the Subject is not authenticated or not authorized to access hidden/secret Services then no custom security token is generated. Based on the information stored in custom security token, the service provider exposes some/all of the hidden/secret services that the Subject is authorized to access. As Subject has no knowledge of custom security token, he cannot know the actual security requirements of these Services and the actual credentials/attributes that cause access to these Services. The custom security token has only limited life time and includes only those minimum credentials and attributes of the Subject that are necessary to access the Service. This model provides two important privacy features to service providers:

- Hidden/Secret Services and their security requirements are not exposed to everyone.
- Even the authorized Subjects don’t know the exact security requirements of hidden/secret Services. As Subjects are given custom security token, they can’t deduce which subset of their credentials/attributes is embedded in the custom security token and is causing access to Services.

TTP can also be used to provide anonymous access to services. As service providers share authentication information of Subjects with TTP, TTP can be used to authenticate Subjects and generate anonymous security tokens. Anonymous security token assert about authorization information of Subjects but removes their identity. Anonymous security token can be used by Subjects to get anonymous access to Services. As anonymous security token has no information regarding the identity of Subjects but only their authorization information, service providers can’t know exactly which Subject is accessing the Service.

Figure 2 shows how anonymous access to a Service is provided. Consider Subject SU wants to access Service SR without disclosing his identity to Service Provider who provides this Service in Domain B. For this, SU requests TTP to generate an anonymous security token based on his identity and access rights. As Service Provider share authentication and authorization information with TTP, TTP verify the identity of SU and generates an anonymous security token. This token has limited lifetime and can be used by SU to get anonymous access to SR. With this token SU can access Service SR without disclosing his identity to Service Provider.

The same setup can be used to access hidden/secret services also. Consider SU wants to access a secret service provided by Domain B. For this SU requests a custom security token from TTP by presenting him his normal security credential. TTP verifies the identity of SU, fetches SU’s attributes from Domain B, prepares a custom security token and sends it to SU. SU can present this custom security token to Domain B to see and access all hidden/secret services that he is authorized to access.

Privacy relationship: A Privacy relationship can be represented as PR = (SUa, DOa, [R], [PU], [ACT], [CO], [OBJ], [SUB], DOb) i.e., Subject SUa of Domain DOa authorizes the set of Subjects [SUB] of Domain DOb to perform any of the actions specified in Action list [ACT] on set of Resources [R] for any of the purpose specified in Purposes set [PU] only if conditions specified in Conditions set [CO] are satisfied and obligations [OBJ] are performed. Notation [x] indicates the set of objects of type x. To specify more fine grained privacy relationship, PR can also be represented as (SUa, DOa, R, PU, ACT, CO,
OB, SUb, DOb) i.e Subject SU a of Domain DOa authorizes Subject SUb of Domain DOb to perform Action ACT on Resource R for Purposes PU only if condition CO is satisfied and obligation OB is performed. Privacy relationships are used by authorization engine to provide privacy based access to Subject’s Resources.

The following sequence of steps can be performed to provide privacy based access to a Service/Resource when a Subject requests it for some Purpose.

**Step 1:** Extract security credentials of Subject from request and verify the identity. If the Subject authentication fails then quit otherwise go to Step 2.

**Step 2:** Determine whether Subject conforms to Privacy Policy (PP) associated with that Service/Resource. For this, perform the following:

- Get the list of authorized Subjects ([SU]) that can access the requested Service/Resource.
- If Subject’s identity does not appear in the list of authorized Subjects then quit otherwise do the following.
- Obtain the purpose set [PU] associated with the requested Service/Resource for which it can be accessed.
- Extract the purpose of access from access request.
- If the purpose for which Subject wants to access the Service/Resource is not listed in the set of purposes ([PU]) then quit otherwise perform the following:
- If the access action specified in the Subject’s access request is not listed in the set of actions allowed ([ACT]) on that Service/Resource for that particular purpose then quit otherwise perform the following.
- Obtain privacy statements and conditions associated with that Service/Resource, the Subject and Resource attributes from the policy and attribute database.
- Check whether the conditions specified in [CO] set are satisfied. If not then deny access otherwise proceed next.
- Check whether the obligations specified in the [OB] set can be performed. If not, deny access otherwise perform Step 3.

**Step 3:** Provide access of Service/Resource to Subject.

**Step 4:** Perform obligations specified in the [OB] set.

**Step 5:** Log access actions in log tables to address auditing and accounting requirements.

The above sequence of steps guarantee that if a Service is exposing Subject’s private information then that information can be accessed by other Subjects/Service only if they are authorized and the purpose for which they are accessing the information matches with the purpose for which the information was actually sent by the Subject to the Service Provider. Privacy Controller can be used to make sure for Subjects that Service Provider is providing privacy based access to private information according to the policies that they have established initially. Thus Subjects can be sure that their private information cannot be misused/accessed inadequately from Service Provider’s database.

**Privacy based authorization framework:** As defined earlier, Privacy based authorization system is a system that grants specific type of access to specific requesters based on their authentication, what services/resources they are accessing, current state of the system and their conformation to established privacy policies. It is a detailed description of all aspects of a system that relate to privacy. Authorization in grid and web services environment needs to be flexible and scalable to support multiple security policies like authentication policies, authorization policies, privacy policies etc. (Lang et al., 2006). In order to provide privacy based access control, we have implemented XACML based Authorization Framework. Figure 3 shows various components of the model. It also shows how the proposed privacy model fits into the framework.

As shown in Fig. 3, authorization request from Subject SU is first intercepted by PEP (Policy Enforcement Point). PEP constructs an authorization decision query and passes it to authorization handler. The result of this query determines if the request is to be granted/deny access to requested Service/Resource. The authorization decision query has details about the identity of the Subject, the Service requested and the purpose for which Service is requested. Authorization Engine passes this information to PDP (Policy Decision Point). The Policy is retrieved by PDP from PRP (Policy Retrieval Point). If the policy information is not available at PRP, it may be retrieved from Policy Store. The Policies are written by administrator using PAP (Policy Administration Point). PIP (Policy Information Point) is used by Authorization Engine to retrieve attributes of Resources, Subjects and Environment. Privacy Handler provides privacy based access control information (as explained in previous Section) to Authorization Engine. After receiving this information, Authorization Engine prepares a final result and passes it to PEP. If Subject conforms to established privacy and other security policies, PEP grants access of
Fig. 3: Privacy based Authorization Framework

Service/Resource to Subject, otherwise the access is denied. After access, the obligation Service, if any, is also executed by PEP.

IMPLEMENTATION DETAILS

For implementation, we are making use of web services security specifications supported by WSE 3.0. Today the world is witnessing the convergence of grid and web services. The two (grid and web) started far apart in specifications, technology and applications but now they are converging into a common set of standards and specifications. The security requirements of grid services overlap deeply with the security requirements of web services as grid services are stateful web services. Web services security specifications consisting of WS-Security (Atkinson et al., 2002), WS-Trust (Anderson et al., 2005b), WS-SecureConversation (Anderson et al., 2005a) etc. have been submitted to OASIS by IBM, Microsoft (Anonymous, 2002) and other leading organizations. These specifications address security requirements like how to associate security tokens with messages (WS-Security), how to express constraints and requirements of a web service (WS-Policy), how to request and issue security tokens to establish trust (WS-Trust and WS-Federation), how to establish and share security contexts (WS-SecureConversation) etc. We are making use of these specifications, supported by WSE 3.0, to implement Privacy based Authorization Framework. Some of the functions defined and used in the prototype implementation are:

AnonymousSecurityCredential getAnonymousCredential (SecurityCredential sc): to retrieve Anonymous Security Credential from TTP.

CustomSecurityCredential getCustomCredential (SecurityCredential sc): to retrieve Custom Security Credential from TTP.

sendRequest(SecurityCredential sc, ServiceRequest sr): to send a request to access a service.

getAuthorizedSubjects(Service sr, Purpose pu): to get list of authorized Subjects that can access Service sr for Purpose pu.

getAllowedActions(Service sr, Purpose pu): to get a list of actions that can be performed on Service sr for Purpose pu.

Following are some of the examples illustrating how privacy policies are represented:

Policy 1: I authorize marketing people of organization to use my telephone number to inform me about new offers only: (I, Home, telephone-no, inform-new-offer, read, null, null, marketing-people, organization).
Policy 2: Banks authorize owners of accounts to change their personal information any time if their age is greater than 18. (Manager, Bank, personal-information, null, read/write, age>18, null, owner-account, Home).

Policy 3: ABC-Travel agency will share personal information of users with government only if ordered by court and users will be notified. (Manager, ABC-Travel, personal-info-users, null, read, order-by-court, notify, officials, Government).

In the prototype implementation we have created 50 Domains with users ranging from 10 to 30 in each Domain. All the Domains have more than 5 service providers that provide different Services/Resources to other Domains. Database contains privacy policies established among Subjects and Services of different Domains. While accessing data services, users mark their data with privacy index according to the sensitivity of information being sent. The privacy policies are exposed by Services in the environment. After agreement with users the purposes, conditions, actions and obligations (as defined by Privacy Relationship) are stored in the database. Other access rules and management policies are also stored in the database using XACML.

All the related information is exchanged as SOAP messages. We have used WSE 3.0 to construct SOAP messages. While constructing these messages, WS-Security information is embedded to handle encryption and signature requirements. SOAP messages make use of WS-Security and related specifications for security token exchange and to address security requirements like confidentiality and integrity. At the target Domain, SOAP message is analyzed and privacy based Authorization Framework is invoked to check Subjects credentials and other attributes against established privacy and other security policies, based on which, access to Service/Resource is either granted/denied. During all these steps the relevant information is stored in log tables also to address auditing and accounting requirements.

The realized prototype implementation is able to meet identified security requirements and privacy policies. This suggests that the approach is workable and the proposed framework can be used to provide privacy based access control to distributed services like grid and web services.

CONCLUSION, SUMMARY AND FUTURE PLANS

The proposed framework provides privacy based access control to distributed services like grid and web services. For implementation we are making use of web services standards and specifications. Currently we have prototype implementation. In future we are planning to use this framework in some real environment. Besides privacy, trust among different entities is also a major concern. We are in the process of defining a trust model also and integrating it with the proposed privacy based Authorization Framework.

REFERENCES

http://www.w3.org/TR/ws-arch/.
http://www.isr.uci.edu/pep05/papers/PEPP.pdf.


